

16/c  
JH  
2-28-02

## APPENDIX

1. A processing agent to transfer data of a predetermined data line length in an external transaction, the agent comprising an internal cache having a plurality of cache entries, each entry sized to store multiple data line lengths of data.
2. The processing agent of claim 1, wherein the cache entries include a tag portion adapted to store address information.
3. The processing agent of claim 2, wherein the internal cache further comprises match detection logic for the tag portions, and control logic provided in communication with the match detection logic.
4. The processing agent of claim 1, wherein the cache entries include a cache coherency state field in association with each data line length of data.
5. The agent of claim 1, further comprising a transaction queue having a plurality of queue entries, the queue entries including a primary entry adapted to store address information and status information of a first external transaction and a secondary entry adapted to store status information of a second external transaction.
6. The agent of claim 5, wherein the status information of the first external transaction includes a field representing whether the first external transaction is part of a multiple transaction sequence.
7. The agent of claim 5, wherein the total number of primary and secondary entries equals the multiple number of data line lengths provided in the cache entries.
8. A processing agent, comprising a transaction queue having a plurality of queue entries, the queue entries each further comprising:
  - a primary sub entry including an address portion and status portion, the status portion provided for a first external transaction of the agent, and

a secondary sub entry including a status portion provided for a second external transaction;

wherein the first and second external transactions relate to the same address portion.

9. The transaction queue of claim 8, wherein the status portion of the primary entry includes a field representing whether the first transaction is part of a multiple transaction sequence.

10. The transaction queue of claim 8, further comprising control logic adapted to cycle through the queue entries and post transactions therefrom.

11. A processing agent, comprising:

an internal cache having cache entries each sized to store multiple data lines, and

a transaction queue system to post external transactions, each external transaction related to a single data line,

wherein the internal cache and the transaction queue system each receive data requests on a common input.

12. The processing agent of claim 11, wherein the internal cache and the transaction queue system communicate by signal lines.

13. The processing agent of claim 12, wherein the signal lines include a cache hit signal line and a tag hit signal line.

14. The processing agent of claim 11, wherein the transaction queue system comprises a plurality of queue entries, each queue entry comprising:

a primary entry including an address portion and status portion, the status portion provided for a first external transaction of the agent, and

a secondary entry including a status portion provided for a second external transaction.

15. The transaction queue of claim 14, wherein the status portion of the primary entry includes a field representing whether the first transaction is part of a multiple transaction sequence.

16. The transaction queue of claim 14, further comprising control logic adapted to cycle through the queue entries and post transactions therefrom.
17. A method of processing a data request within a processing agent comprising:  
posting the data request internally within the agent,  
determining whether the request hit the cache,  
when the request misses the cache, posting a sequence of external transactions to fill a cache line with data associated with the data request;  
wherein each cache line is sized to store multiple data line lengths of data.
18. The method of claim 17, wherein the determining step includes:  
comparing address information of the data request with tags stored in the internal cache,  
and  
identifying a cache miss when the address information does not match any stored tag.
19. The method of claim 18, wherein the determining step further includes:  
when address information matches a stored tag, reading cache coherency state information associated with the requested data, and  
identifying a cache miss when the cache coherency state information is invalid for a request type of the data request.
20. The method of claim 17, further comprising, when the request hits the cache:  
determining whether the request hits a tag stored in the cache, and  
if so, generating a single external transaction to read the requested data into the agent.
21. The method of claim 20, wherein the second determining step includes:  
comparing address information of the data request with tags stored in the internal cache,  
and  
identifying a tag hit when the address information matches a stored tag.
22. An agent comprising a transaction queue, the transaction queue further comprising a plurality of queue entries, each queue entry comprising:  
an address field,

(cont'd.)  
a first status field to store data associated with a first transaction based on the address field, and  
another status field to store data associated with another transaction based on the address field.

23. A method of processing a data request within a processing agent comprising:  
posting the data request internally within the agent,  
determining whether the request hit the cache,  
when the request misses the cache, posting a series of external transactions from within the agent to fill a cache line with data associated with the data request, the external transactions directed to a data-line-sized data item identified by an address of the data request and to at least one other data-line-sized data item adjacent to the first data item;  
wherein each cache line is sized to store multiple data line lengths of data.
24. A processing agent, comprising:  
an internal cache having a plurality of cache lines, each cache line including:  
a tag portion storing address information, and  
a plurality of cache entries, each cache entry sized to store one data line length of data;  
wherein the processing agent, in response to a multiple transaction signal, posts a series of external transactions related to the address information, each of said external transactions filling one of said cache entries in the cache line.
25. The processing agent of claim 1, wherein "data line" refers to the maximum amount of data that can be transferred in a single bus transaction.
26. The processing agent of claim 11, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.
27. The method of claim 17, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.

28. The method of claim 23, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.

29. The processing agent of claim 24, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.

2. The processing agent of claim 1, wherein the cache entries include a tag portion adapted to store address information.

3. The processing agent of claim 2, wherein the internal cache further comprises match detection logic for the tag portions, and control logic provided in communication with the match detection logic.

4. The processing agent of claim 1, wherein the cache entries include a cache coherency state field in association with each data line length of data.

5. The agent of claim 1, further comprising a transaction queue having a plurality of queue entries, the queue entries including a primary entry adapted to store address information and status information of a first external transaction and a secondary entry adapted to store status information of a second external transaction.

6. (Amended) The agent of claim 4<sup>5</sup>, wherein the status information of the first external transaction includes a field representing whether the first external transaction is part of a multiple transaction sequence.

7. (Amended) The agent of claim 4<sup>5</sup>, wherein the total number of primary and secondary entries equals the multiple number of data line lengths provided in the cache entries.

8. (Amended) A processing agent, comprising a transaction queue having a plurality of a queue entries, the queue entries each further comprising:

a primary sub entry including an address portion and status portion, the status portion provided for a first external transaction of the agent, and

a secondary sub entry including a status portion provided for a second external transaction;

wherein the first and second external transactions relate to the same address portion.

9. The transaction queue of claim 8, wherein the status portion of the primary entry includes a field representing whether the first transaction is part of a multiple transaction sequence.

10. The transaction queue of claim 8, further comprising control logic adapted to cycle through the queue entries and post transactions therefrom.

11. A processing agent, comprising:

an internal cache having cache entries each sized to store multiple data lines, and  
a transaction queue system to post external transactions, each external transaction related to a single data line,

wherein the internal cache and the transaction queue system each receive data requests on a common input.

12. The processing agent of claim 11, wherein the internal cache and the transaction queue system communicate by signal lines.

13. The processing agent of claim 12, wherein the signals lines include a cache hit signal line and a tag hit signal line.

14. The processing agent of claim 11, wherein the transaction queue system comprises a plurality of queue entries, each queue entry comprising:

a primary entry including an address portion and status portion, the status portion provided for a first external transaction of the agent, and

a secondary entry including a status portion provided for a second external transaction.

15. The transaction queue of claim 14, wherein the status portion of the primary entry includes a field representing whether the first transaction is part of a multiple transaction sequence.

16. The transaction queue of claim 14, further comprising control logic adapted to cycle through the queue entries and post transactions therefrom.

17. (Amended) A method of processing a data request within a processing agent comprising:

posting the data request internally within the agent,

determining whether the request hit the cache,

when the request misses the cache, posting a sequence of external transactions to fill a cache line with data associated with the data request;

wherein each cache line is sized to store multiple data line lengths of data.

C/ 18. The method of claim 17, wherein the determining step includes:  
comparing address information of the data request with tags stored in the internal cache,  
and  
identifying a cache miss when the address information does not match any stored tag.

19. The method of claim 18, wherein the determining step further includes:  
when address information matches a stored tag, reading cache coherency state  
information associated with the requested data, and  
identifying a cache miss when the cache coherency state information is invalid for a  
request type of the data request.

20. The method of claim 17, further comprising, when the request hits the cache:  
determining whether the request hits a tag stored in the cache, and  
if so, generating a single external transaction to read the requested data into the agent.

21. The method of claim 20, wherein the second determining step includes:  
comparing address information of the data request with tags stored in the internal cache,  
and  
identifying a tag hit when the address information matches a stored tag.

22. An agent comprising a transaction queue, the transaction queue further comprising a plurality of queue entries, each queue entry comprising:  
an address field,  
a first status field to store data associated with a first transaction based on the address field, and  
another status field to store data associated with another transaction based on the address field.

23. (Amended) A method of processing a data request within a processing agent comprising:



posting the data request internally within the agent,  
determining whether the request hit the cache,

when the request misses the cache, posting a series of external transactions from within the agent to fill a cache line with data associated with the data request, the external transactions directed to a data-line-sized data item identified by an address of the data request and to at least one other data-line-sized data item adjacent to the first data item;

wherein each cache line is sized to store multiple data line lengths of data.

Please add the following new claim(s).

24. A processing agent, comprising:  
an internal cache having a plurality of cache lines, each cache line including:  
a tag portion storing address information, and  
a plurality of cache entries, each cache entry sized to store one data line length of data;  
wherein the processing agent, in response to a multiple transaction signal, posts a series of external transactions related to the address information, each of said external transactions filling one of said cache entries in the cache line.
25. The processing agent of claim 1, wherein "data line" refers to the maximum amount of data that can be transferred in a single bus transaction.
26. The processing agent of claim 11, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.
27. The method of claim 17, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.
28. The method of claim 23, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.
29. The processing agent of claim 24, wherein "data line" corresponds to the maximum amount of data that can be transferred in a single bus transaction.